Database Architecture

Duration: 3 hours

Detailed Syllabus

1.2. Database Architecture

1.2.1. Components of a Database Management System:

Data Dictionary (importance, contents)
Meta data
Data security and integrity
Concurrent access
User-oriented data query and reporting
Application development facilities

1.2.2. Database Systems:

ANSI/SPARC Three-level Architecture
Conceptual model, Logical model, Physical model
External view, Conceptual view, Internal view of data

Detailed Syllabus Contd.

1.2.3. Data specification and access mechanisms:

Data Definition Language (DDL)
Sub-Schema DDL (SDDL)
Data Manipulation Language (DML)
End users, Database Administrator
Functions, Capabilities of DBMS
Advantages and disadvantages

Data Dictionary/System Catalog

- **A subsystem that keeps track of the definitions of data items in the database which includes**
  - Elementary-level data items (fields/attributes),
  - group and record-level data structures, and
  - files or relational tables.

Meta Data

- Data that describe the properties or characteristics of other data.
- Some of these properties include the name of the data item, data type, length, minimum and maximum allowable values (where appropriate), rules or constraints and a brief description of each data item.
- Metadata allow database designers and users to understand what data exist, what the data mean.
- Data without clear meaning can be confusing, misinterpreted or erroneous.

Meta Data

- E.g. Employee

Name	Type	Length	Min	Max	Description
EmpNo	Number	9			Employee No.
Name	Character	30			Employee Name
Dept	Character	10			Dept. No.
Salary	Number	8	5000	60000	Employee Salary

- Employee No. (ID) unique

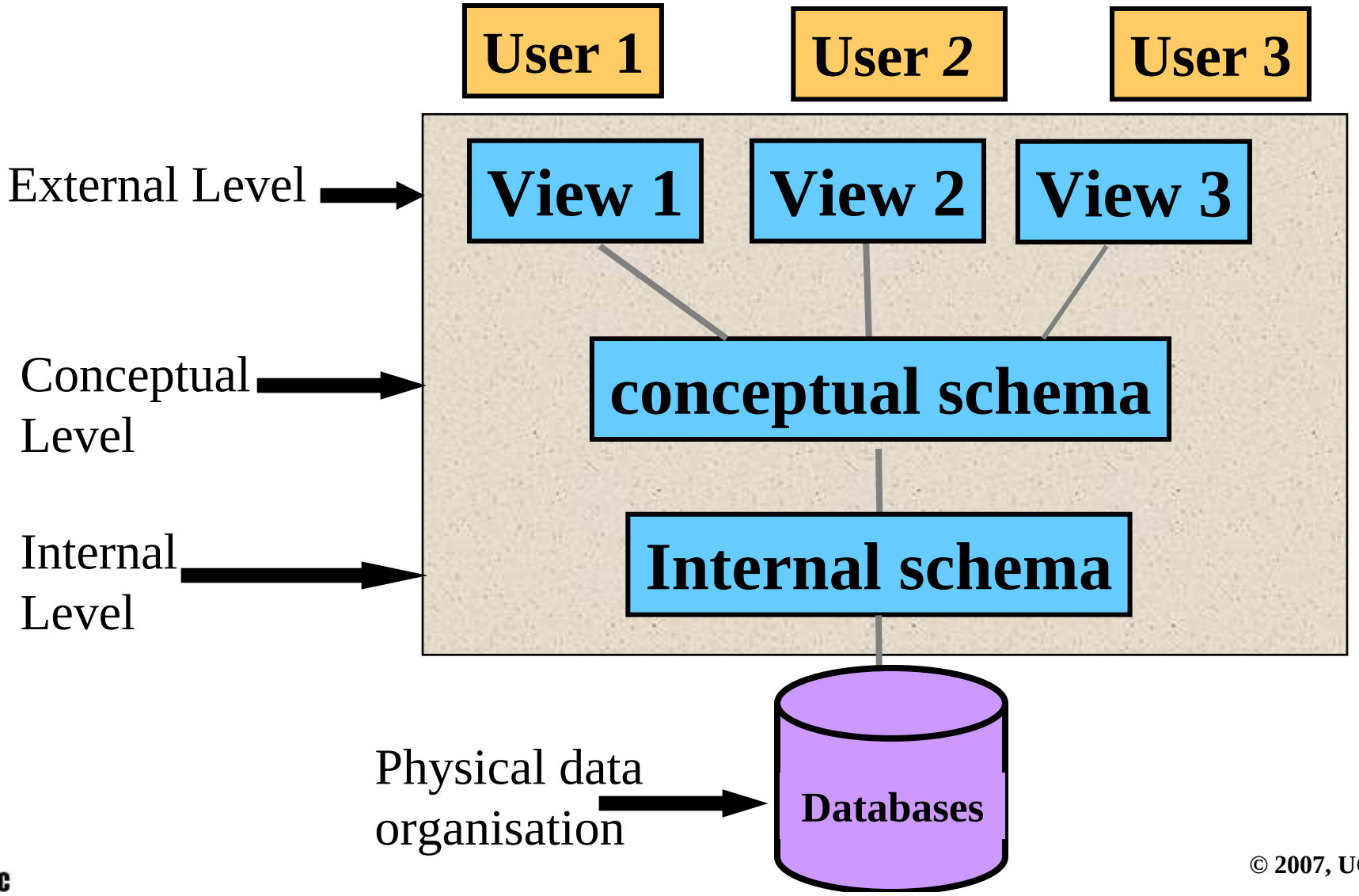
Three-Level Architecture

- All users should be able to access same data but have a customized view of the data.
- A user's view is immune to changes made in other views.
- Users should not need to know physical database storage details (e.g. indexing or hashing).

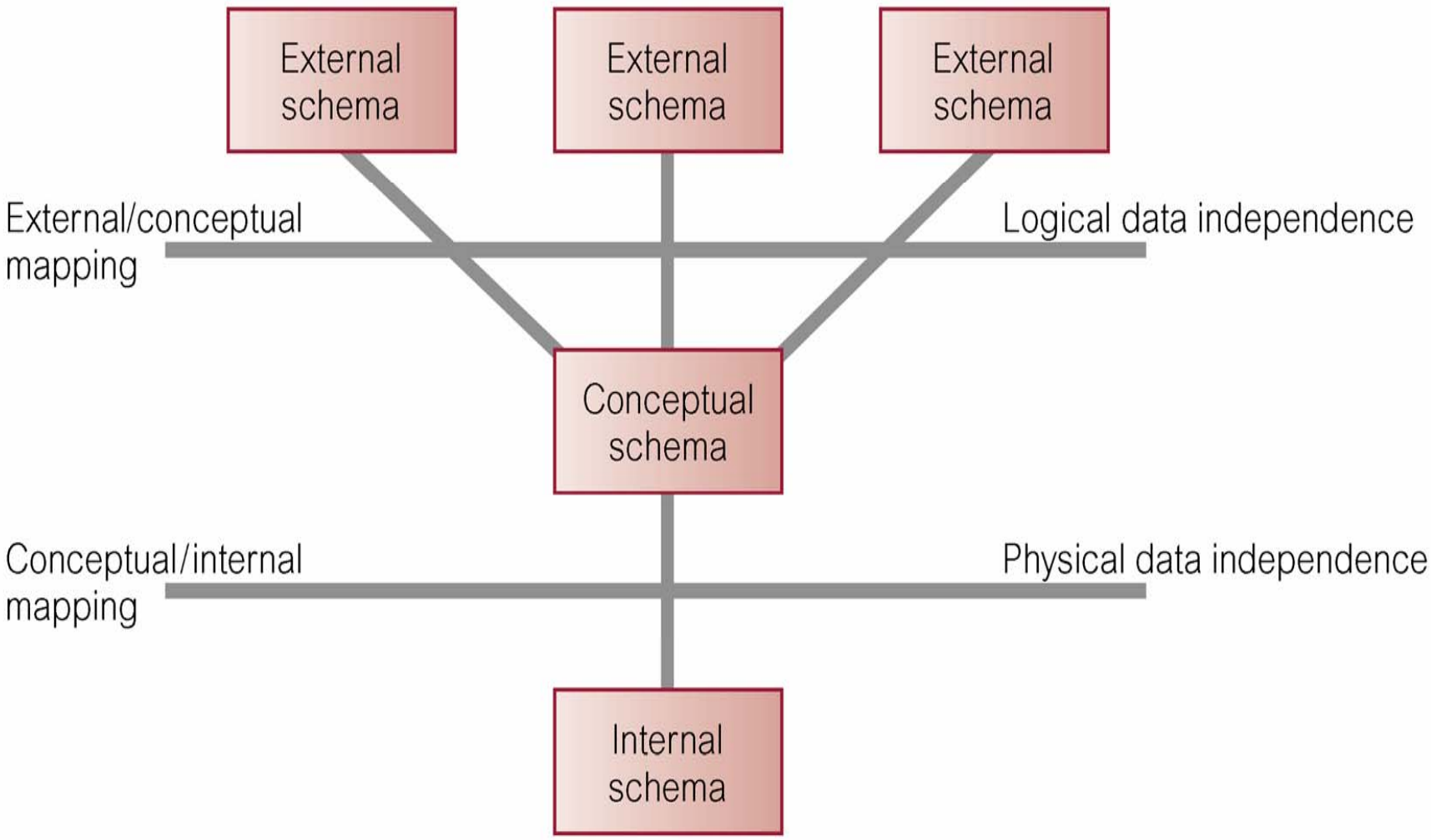
3 Level ANSI/SPARC Architecture

- External level
  - User's view of the database.
- Conceptual level
  - Describes what data is stored in the database and the relationships among the data.
- Internal
  - Describes how the data is stored in the database.

Three-level ANSI/SPARC architecture



Data Independence and the ANSI-SPARC Three-Level Architecture



Internal Level

- **The physical representation of the database on the computer to achieve optimal runtime performance and storage space utilization.**

  - **Covers data structures and file organisations used to store data on the storage device.**
  - **Storage space allocation for data and indexes.**

Conceptual Level

- This level contains the logical structure of the entire database. Provides a complete view of the data requirements of the organization that is independent of any storage considerations.

- The conceptual level represents:
  - All entities, their attributes and their relationships
  - The constraints on the data
  - Security and integrity information

External Level

- Describes the part of the database that is relevant to the user.
- The external view include only the entities, attributes or relationships in the ‘real world’ that the user is interested in.
- Different views have different representations of the same data.

External Level

- **External Views Allow to**
  - **hide unauthorised data**
    - **e.g. *salary, dob***
  - **provide user view**
    - **e.g. view employee *name, designation, department* data taken from *employee* and *department* files**
  - **derive new attributes**
    - **e.g. *age* derived from *dob* or *nid***

External Level

- **External Views Allow to**
  - **change unit of measurement**
    - **e.g. show *age* in years or months**
  - **define security levels**
    - **e.g. update access to *employee* file read-only to *department* file**

Objectives of Three-Level Architecture

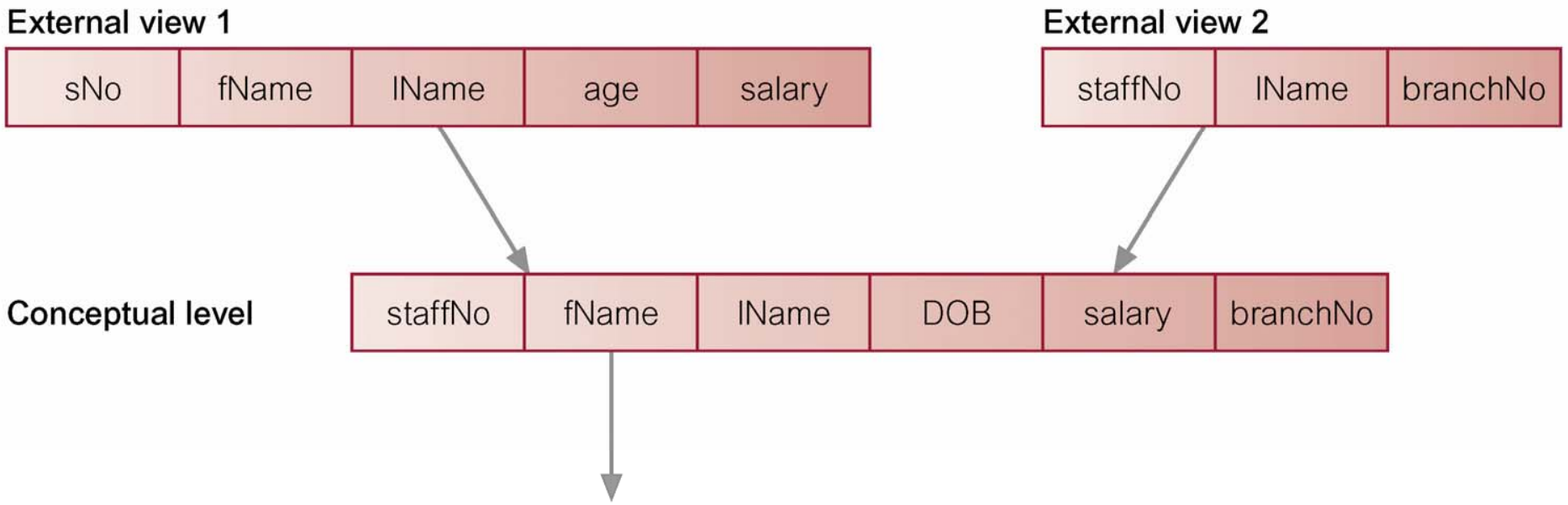
- DBA should be able to change database storage structures without affecting the users' views.

- DBA should be able to change conceptual structure of database without affecting all users.

Physical Level

- Managed by the operating system under the direction of the DBMS.
- Consist of items only the OS knows.

Differences between Three Levels of ANSI-SPARC Architecture



Internal level

```
struct STAFF {
        int staffNo;
        int branchNo;
        char fName [15];
        char lName [15];
        struct date dateOf Birth;
        float salary;
        struct STAFF *next;                    /* pointer to next Staff record */
};
index staffNo; index branchNo;                 /* define indexes for staff */
```

Mapping between Levels

- **DBMS map or translate from one level to another.**
  - **External ⇔ Conceptual**

  External schema is related to the conceptual schema
  - **Conceptual ⇔ Internal**

  Conceptual schema is related to the internal schema.

Data Independence

- **Logical Data Independence**
  - **Refers to immunity of external schemas to changes in conceptual schema.**
  - **Conceptual schema changes (e.g. addition/removal of entities).**
  - **Should not require changes to external schema or rewrites of application programs.**

Data Independence

- **Physical Data Independence**
  - **Refers to immunity of conceptual schema to changes in the internal schema.**
  - **Internal schema changes (e.g. using different file organizations, storage structures/devices).**
  - **Should not require change to conceptual or external schemas.**

Database Approach

- **Data definition language (DDL).**
  - **Permits specification of data types, structures and any data constraints.**
  - **All specifications are stored in the database.**
- **Data manipulation language (DML).**
  - **General enquiry facility (query language) of the data.**

Database Approach

- **Controlled access to database may include:**
  - **A security system.**
  - **An integrity system.**
  - **A concurrency control system.**
  - **A recovery control system.**
- **A view mechanism.**
  - **Provides users with only the data they want or need to use.**

Views

- Allows each user to have his or her own view of the database.
- A view is essentially some subset of the database.
- Benefits include:
  - Provide a level of security;
  - Provide a mechanism to customize the appearance of the database;

Database Languages

- **Data Definition Language (DDL)**
  - **Allows the DBA or user to describe and name entities, attributes, and relationships required for the application**
  - **plus any associated integrity and security constraints.**

Database Languages

- **Data Manipulation Language (DML)**
  - **Provides basic data manipulation operations on data held in the database.**
- **Non-Procedural DML**
  - **allows user to state what data is needed rather than how it is to be retrieved.**
- **Procedural DML**
  - **allows user to tell system exactly how to manipulate data.**

Database Applications

Databases range from those for a single user with a desktop computer to those on mainframe computers with thousands of users.

- Personal databases
- Workgroup databases
- Departmental databases
- Enterprise databases

Personal databases

Designed to support one user with a stand alone PC.

E.g. a sales person keeping track of this customer information with contact details.

Workgroup databases

A relatively small team of people (less than 25) who collaborate on the same project or application.

E.g. a team of engineering designers maintain versions of the artifact that they design.

Departmental databases

A department is a functional unit of an organisation. It is larger than a workgroup.

Department databases are designed to support the various functions and activities of a department.

E.g. a personnel database that is designed to track data concerning employees, jobs, skills and job assignments.

Enterprise databases

An enterprise is one whose scope is the entire organisation or enterprise.

Such databases are intended to support organisation-wide operations and decision making.

E.g. a large health care organisation that operates a group of medical centre's including hospitals, clinics and nursing homes.

Enterprise databases

An enterprise database does support information needs from many departments. The most important type of enterprise database today is called a data warehouse.

- **Data warehouse**
  - **An integrated decision support database whose content is derived from the various operational databases.**

Database Approach -Advantages

- Improved maintenance through program-data independence
- Minimal data redundancy
- Improved data consistency
- Improved data sharing
- Increased productivity

Advantages

- Enforcement of standards
- Improved data integrity
- Improved data accessibility and responsiveness
- Improved security
- Increased concurrency

Improved maintenance through Program-Data/Data Independence

- **The separation of data descriptions (metadata) from the application programs that use the data.**

  This simplifies database application maintenance.

- **In the database approach data descriptions are stored in a central location called the data dictionary. This property allows an organisation's data to change and evolve (within limits) without changing the application program that process the data.**

Minimal Data Redundancy

- **Data files are integrated into a single, logical structure. Each primary fact is recorded (ideally) in only one place in the database.**
- **E.g. Employee data not with the payroll and benefit files.**

Note: Data redundancy is not eliminated entirely. Some data items will appear in more than one place (e.g. employee no.) to represent the relationship with others.

Improved Data Consistency

- By eliminating (or controlling) data redundancy, we greatly reduce the opportunities for inconsistency.

  E.g. employee address is stored only once and hence we cannot have disagreement on the stored values.
- Also, updating data values is greatly simplified and have avoided the wasted storage space.

Improved Data Sharing

- **A database is designed as a shared corporate resource and can be shared by all authorised users. In this way more users share more of the data.**

  E.g. employee data common to payroll, benefit applications will be shared among different users.

  New applications can be built on the existing data in the database.

Increased Productivity

- **A major advantage of the database approach is that it greatly reduces the cost and time for developing new business applications.**
  - **Programmer could concentrate on the specific functions required for the new application, without having to worry about design or low-level implementation details; as related data has already been designed and implemented.**
  - **DBMS provides many of the standard functions (e.g. forms and report generations) that the programmer would normally have to write in a file-based application DBMS.**

Enforcement of Standards

- When the database approach is implemented with full management support, the database administration function should be granted single-point authority and responsibility for establishing and enforcing data standards.

- Standards include naming conventions, data quality standards and uniform procedures for accessing, updating and protecting data.

Improved Data Integrity

- **Integrity can be expressed in terms of *constraints*, which are consistency rules that the database is not permitted to violate.**

Eg: A member of staff's salary cannot be greater than 60,000.

Improved Data Accessibility and Responsiveness

- **With relational database, end users without programming experience can often retrieve and display data, even when it crosses traditional departmental boundaries.**

- **English-like query language SQL and query tools such as Query-By-Example provide such facilities.**

Improved Security

- DBMS can be used to enforce database security. This may take the form of user names and passwords to identify people authorised to use the database.

- The access that the authorised user is allowed on the data can also be restricted by the operation type (retrieval, delete, update, insert).

Increased concurrency

- Many DBMSs allow users to undertake simultaneous operations on the database. The DBMS implements a *concurrency control* mechanism that prevents database accesses from interfering with one another.

Disadvantages of DBMSs

- Complexity
- Size
- Cost of DBMS
- Additional hardware costs
- Cost of conversion
- Performance
- Higher impact of a failure